

CLAIMS

What is claimed is:

1. A method of providing secure information, the method comprising regenerating a new  
5 encryption key with an encryption key, encrypted data, and a hash vector based upon an encryption key.
2. The method of claim 1 wherein the step of regenerating a new encryption key with an  
encryption key, encrypted data, and a hash vector based upon an encryption key comprises  
10 performing byte addition of an encryption key, encrypted data, and a hash vector based upon an encryption key.
3. The method of claim 1 further comprising the step of hashing a hash vector based  
upon an encryption key.  
15
4. The method of claim 3 wherein the step of hashing a hash vector based upon an  
encryption key comprises:  
scanning indexed bytes of an encryption key; and  
using indices and associated values of indices of an encryption key as  
20 indices of two bytes in a hash vector to be swapped.
5. The method of claim 1 wherein the step of regenerating a new encryption key with an  
encryption key, encrypted data, and a hash vector based upon an encryption key comprises:  
selecting a previously encrypted data record; and  
25 regenerating a new encryption key with an encryption key, selected  
encrypted data, and a hash vector based upon an encryption key.

6. The method of claim 5 wherein the step of selecting a previously encrypted data record comprises:

randomly selecting an index from the range  $[1, t-1]$  using a byte of an encryption key as a seed of random generation; and

5 selecting the previously encrypted data record corresponding to the selected index.

7. The method of claim 1 wherein the step of regenerating a new encryption key with an encryption key, encrypted data, and a hash vector based upon an encryption key comprises

10 regenerating a new encryption key with an encryption key, previously encrypted data, a hash vector based upon an encryption key, and a received cipher.

8. A method of providing secure information, the method comprising the steps of:

generating  $n$  encryption keys;

15 encrypting  $n$  tracks of data records with  $n$  tracks of parallel encryption; and

regenerating an encryption key with an encryption key, a hash vector based upon an encryption key, and selected encrypted data.

9. The method of claim 8 wherein the step of regenerating an encryption key with an encryption key, a hash vector based upon an encryption key, and selected encrypted data comprises:

randomly selecting an index from the range  $[1, t-1]$  using a byte of an encryption key as a seed of random generation; and

25 selecting the previously encrypted data record corresponding to the selected index.

10. A method of providing secure information, the method comprising the steps of:  
encrypting a data record with a hash vector based upon an encryption key;  
and  
regenerating an encryption key with an encryption key and encrypted data.

5

11. The method of claim 10 wherein the step of encrypting a data record with a hash vector based upon an encryption key comprises performing a logic operation on a data record and a hash vector based upon an encryption key.

10

12. The method of claim 11 wherein the step of performing a logic operation on a data record and a hash vector based upon an encryption key comprises performing an XOR operation on a data record and a hash vector based upon an encryption key.

15

13. The method of claim 10 further comprising the step of decrypting encrypted data, comprising performing a logic operation on an encrypted data record and a hash vector based upon an encryption key.

20

14. The method of claim 13 wherein the step of performing a logic operation on an encrypted data record and a hash vector based upon an encryption key comprises performing an XOR operation on an encrypted data record and a hash vector based upon an encryption key.

15. A system for providing secure information, the system comprising:  
a source node;  
a destination node;  
5 a data stream created at said source node;  
means for encrypting data of said data stream with a hash vector based upon  
an encryption key; and  
means for regenerating a new encryption key with an encryption key,  
encrypted data, and a hash vector based upon an encryption key.

10 16. A method of authenticating one system node to another system node, the method  
comprising the steps of:

generating an authentication key at a node;  
transmitting the authentication key to another node; and  
15 starting a daemon at each node for regenerating a new authentication key  
with an authentication key, an auxiliary key, and a hash vector based upon an authentication key,  
and maintaining a corresponding number-regeneration-counter at each node.

20 17. The method of claim 16 wherein the step of regenerating a new authentication key  
with an authentication key, an auxiliary key, and a hash vector based upon an authentication key  
comprises performing byte addition of an authentication key, an auxiliary key, and a hash vector  
based upon an authentication key.

25 18. The method of claim 16 further comprising the step of generating an auxiliary key  
from at least one key selected from the group consisting of encryption keys, authentication keys, and  
a hash vector based upon an authentication key.

30 19. The method of claim 18 wherein the step of generating an auxiliary key comprises  
generating an auxiliary key by performing byte addition of an authentication key, an encryption key,  
and a hash vector based upon an authentication key.

20. The method of claim 18 wherein the step of generating an auxiliary key comprises generating an auxiliary key by performing byte addition of two or more authentication keys and a hash vector based upon an authentication key.

5

21. A method of validating data integrity, the method comprising the steps of:

buffering an encryption key and a hash vector based upon an encryption key  
at a source node;

10 encrypting a data record using a hash vector based upon an encryption key  
to yield a cipher record of a first point in time at a source node;

transmitting the encrypted data record to a destination node;

receiving a cipher from a destination node;

decrypting the received cipher from the destination node with a hash vector  
based upon an encryption key of a second point in time; and

15 comparing the decrypted received cipher to a data record.

22. The method of claim 21 further comprising the steps of:

buffering an encryption key and a hash vector based upon an encryption key  
at a destination node;

5 encrypting a data record using a hash vector based upon an encryption key  
to yield a cipher record of a second point in time at a destination node;

transmitting the encrypted data record to a source node;

receiving a cipher from a source node;

decrypting the received cipher from the source node with a hash vector  
based upon an encryption key of a first point in time; and

10 comparing the decrypted received cipher to a data record.

23. A method of synchronizing one node to another node, the method comprising the  
steps of:

15 receiving a request from a first user to communicate with a second user  
along with an authentication key number regeneration count and a hashed value of an authentication  
key number regeneration count;

requesting an authentication key number regeneration count and a hashed  
value of an authentication key number regeneration count from a second user;

20 comparing a central authority authentication key number regeneration count  
to a user authentication key number regeneration count; and

aligning the authentication keys of a user and a central authority node  
according to the comparison.

24. The method of claim 23 wherein the step of receiving a request from a first user to communicate with a second user along with an authentication key number regeneration count and a hashed value of an authentication key number regeneration count comprises receiving a request  
5 from a first user to communicate with a second user along with an authentication key number regeneration count and a hashed value of an authentication key number regeneration count encrypted with a static key.

25. The method of claim 23 wherein the step of requesting an authentication key number  
10 regeneration count and a hashed value of an authentication key number regeneration count from a second user comprises requesting an authentication key number regeneration count and a hashed value of an authentication key number regeneration count encrypted with a static key from a second user.

15 26. The method of claim 23 further comprising the step of authenticating the identity of the first and second user.

27. The method of claim 26 wherein the step of authenticating the identity of the first and second user comprises:

20 generating a nonce at a central authority node;  
encrypting a nonce with a hash vector of an authentication key;  
transmitting an encrypted nonce to a user node;  
decrypting an encrypted nonce at a user node; and  
comparing a decrypted nonce with a nonce.

25

28. The method of claim 27 wherein the step of encrypting a nonce with a hash vector of an authentication key comprises:

- 5                   generating additional authentication keys; and  
                  encrypting a nonce with a hash vector of an additional authentication key.

29. The method of claim 27 further comprising the steps of:

- generating additional authentication keys;  
                  transmitting a nonce encrypted with a hash vector of an additional  
10 authentication key to a central authority;  
                  decrypting an encrypted nonce at a central authority; and  
                  comparing a decrypted nonce with a nonce at a central authority.